

(19)



(11)

EP 2 853 062 B1

(12)

EUROPEAN PATENT SPECIFICATION

(45) Date of publication and mention
of the grant of the patent:

09.05.2018 Bulletin 2018/19

(51) Int Cl.:

H04L 12/26 ^(2006.01)

(86) International application number:

PCT/EP2012/059444

(21) Application number: **12722387.3**

(22) Date of filing: **22.05.2012**

(87) International publication number:

WO 2013/174417 (28.11.2013 Gazette 2013/48)

(54) TIME MEASUREMENT ON A PACKET FLOW IN A PACKET-SWITCHED COMMUNICATION NETWORK

ZEITMESSUNG AUF EINEM PAKETSTROM IN EINEM PAKETVERMITTELTEM
KOMMUNIKATIONSNETZ

MESURE DE TEMPS SUR UN FLUX DE PAQUETS DANS UN RÉSEAU DE COMMUNICATION À
COMMUTATION DE PAQUETS

(84) Designated Contracting States:

**AL AT BE BG CH CY CZ DE DK EE ES FI FR GB
GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO
PL PT RO RS SE SI SK SM TR**

(43) Date of publication of application:

01.04.2015 Bulletin 2015/14

(73) Proprietor: **Telecom Italia S.p.A.**

20123 Milano (IT)

(72) Inventor: **COCIGLIO, Mauro**

I-10148 Torino (IT)

(74) Representative: **Colombo, Stefano Paolo et al**

Marchi & Partners S.r.l.

Via Vittor Pisani, 13

20124 Milano (IT)

(56) References cited:

WO-A1-01/88763

WO-A1-2011/079857

GB-A- 2 422 694

EP 2 853 062 B1

Note: Within nine months of the publication of the mention of the grant of the European patent in the European Patent Bulletin, any person may give notice to the European Patent Office of opposition to that patent, in accordance with the Implementing Regulations. Notice of opposition shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European Patent Convention).

Description**Technical field**

[0001] The present invention relates to the field of communication networks. In particular, the present invention relates to a method for performing a time measurement (in particular, a measurement of delay and/or interarrival jitter), on a packet flow transmitted in a packet-switched communication network. Further, the present invention relates to nodes and computers for communication network configured to implement such a method, and to computer networks comprising such nodes and computers.

Background art

[0002] In a packet-switched communication network, data are transmitted in the form of packets that are routed from a source node to a destination node through possible intermediate nodes. Exemplary packet-switched networks are IP (Internet Protocol) networks, Ethernet networks and MPLS (Multi-Protocol Label Switching) networks.

[0003] Packets not always reach the destination node, i.e. they may be lost during transmission through the network. Loss of packets is due to different reasons. For instance, a node or link may fail, thereby causing a total packet loss until the failure is either bypassed or fixed. Alternatively, packets may be discarded by a node due to a congestion of its ports. Besides, packets may be discarded by a node since they contain bit errors. In any case, when providing a service by transmitting data through a packet-switched network, the rate of packets lost during transmission affects the quality of service (QoS) of that service.

[0004] Besides, a packet is transmitted at a transmission time by the source node and is received at a reception time by the destination node. The time elapsing between transmission time and reception time is typically called "one way delay" (or, briefly, "delay"). The delay of a packet is given by the following equation:

$$D = T_{rx} - T_{tx}, \quad [1]$$

where T_{rx} is the transmission time and T_{tx} is the reception time of the packet.

[0005] The delay of a packet mainly depends on the number of possible intermediate nodes crossed by the packet from source to destination, the permanence time of the packet at each node and the propagation time along the links. Since the packets are routed hop-by-hop by each node, both the number of possible intermediate nodes crossed by packets and the permanence time of packets at each node are unpredictable. Accordingly, the delay of a packet is almost unpredictable.

[0006] Besides, packets of a same packet flow may have different delays. The difference in the delays of two packets of a same data flow is termed "interarrival jitter". In particular, if T_{tx1} and T_{tx2} are the transmission times for a first packet and a second packet, and T_{rx1} and T_{rx2} are the reception times for the first packet and the second packet, the interarrival jitter may be expressed as:

$$J = (T_{rx1} - T_{rx2}) - (T_{tx1} - T_{tx2}). \quad [2]$$

[0007] When a communication service (in particular, a real-time voice or data service such as call, conference call, video conference, etc.) is provided by means of a packet-switched network, the delay and interarrival jitter of the packet flow carrying the service strongly affect the quality of service (QoS) perceived by the end users of the service. Therefore, measuring delay/interarrival jitter of packet flows in a communication network is of particular interest for network operators.

[0008] WO 2010/072251 (in the name of the same Applicant) discloses a method for measuring data loss of a data flow transmitted through a communication network from a transmitting node to a receiving node. Before transmitting the data units of the data flow, the transmitting node marks each data unit for dividing the data flow in blocks. In particular, the transmitting node marks each data unit by setting a bit of its header to "1" or "0". The marking results in a sequence of blocks, where blocks of data units marked with "1" alternate in time with blocks of data units marked with "0". The blocks may have a same duration termed "block period" T_b (e.g. 5 minutes). Further, the transmitting node increases by one a first counter $C1$ each time a data unit marked by "1" is transmitted, and a second counter $C0$ each time a data unit marked by "0" is transmitted. The marked data units are then received at the receiving node. Each time the receiving node receives a data unit, it checks its marking, increases a third counter $C'1$ if the marking is "1" and increases a fourth counter $C'0$ if the marking is "0". The values of the counters $C1$, $C0$, $C'1$ and $C'0$ are periodically detected and the detected values are used for calculating the data loss at each block period.

[0009] WO 2011/079857 (in the name of the same Applicant) discloses a method for performing a time measurement

(in particular, for measuring delay and/or interarrival jitter) on a data flow transmitted from a transmitting node to a receiving node of a communication network. According to WO 2011/079857, at the transmitting node, in addition to the above marking operation disclosed by WO 2010/072251, a transmission timestamp is also generated at each block period, which indicates the time at which a predetermined data unit of the current block (for instance, the first data unit of the current block) is transmitted. At the receiving node, a reception timestamp is generated at each block period, which indicates the time at which the predetermined data unit of the current block is received. At each block period, the transmission timestamp and the reception timestamp are used for calculating the delay of the predefined data unit of the current block.

Summary of the invention

[0010] The Applicant has perceived the need to improve the solution for performing a time measurement described by WO 2011/079857.

[0011] Indeed the above solution provides, for each block period, a single delay measurement referred to a sample packet (namely, the predetermined data unit). Such delay measurement does not provide any indication of the delays undergone by the other packets transmitted during the same block period, which may be very different from the delay undergone by the sample packet.

[0012] Further, in case the sample packet of a given block period is lost before it reaches the receiving node, no significant delay measurement may be calculated for that block period. Also in case a reception sequence error occurs in a block period (namely, the packets are received in an order different from the order in which they were transmitted) which involves the sample packet, no significant delay measurement may be calculated for that block period.

[0013] In view of the above, the Applicant has tackled the problem of providing a method for performing a time measurement on a packet flow transmitted in a packet-switched communication network, which provides a result indicative of the behaviour of all the transmitted packets and which is more robust against possible packet loss and reception sequence errors.

[0014] In the following description and in the claims, the expression "performing a time measurement" will designate an operation of measuring:

- a delay induced on a packet by transmission from the first node to the second node; and/or
- an interarrival jitter induced on a pair of packets by transmission from the first node to the second node.

[0015] Besides, in the following description and in the claims, the expression "marking a packet" will designate an operation of setting a feature of the packet to a predefined marking value, in particular to one of at least two alternative marking values. For instance, the operation of marking a packet may comprise the operation of setting one or more bits of the packet (e.g. one bit or a bit sequence of its header) to one of at least two predefined alternative marking values, the operation of setting its frequency or its phase to one of at least two predefined alternative marking values, and so on.

[0016] According to a first aspect, the present invention provides a method for performing a time measurement on a packet flow to be transmitted from a first node to a second node of a communication network, the packet flow being divided, upon its transmission, in first packets transmitted during first block periods and second packets transmitted during second block periods, the first block periods alternating in time with the second block periods, the packet flow comprising at least two packets transmitted by the first node during a block period amongst the first block periods or the second block periods, the method comprising:

- a) calculating a medium transmission time parameter for the block period, indicative of an average of at least two transmission time parameters relating to the at least two packets, wherein the medium transmission time parameter comprises a medium transmission timestamp calculated as an average of at least two transmission timestamps generated upon transmission of the at least two packets;
- b) calculating a medium reception time parameter for the block period, indicative of an average of at least two reception time parameters relating to the at least two packets, wherein the medium reception time parameter comprises a medium reception timestamp calculated as an average of at least two reception timestamps (generated upon reception of the at least two packets; and
- c) calculating a medium time measurement for the block period, indicative of an average performance of the packet flow during the block period using the medium transmission time parameter and the medium reception time parameter, wherein the medium time measurement comprises a medium delay calculated as a difference between medium reception timestamp and medium transmission timestamp.

[0017] Preferably:

- at step a), the medium transmission time parameter comprises a medium one-point transmission jitter calculated as a difference between a last-packet transmission timestamp indicative of a transmission time of a last packet of the at least two packets and a first-packet transmission timestamp indicative of a transmission time of a first packet of the at least two packets, divided by a first number of packets transmitted by the first node during the block period decreased by 1;
- at step b), the medium reception time parameter comprises a medium one-point reception jitter calculated as a difference between a last-packet reception timestamp indicative of a reception time of the last packet of the at least two packets and a first-packet reception timestamp indicative of a reception time of the first packet of the at least two packets, divided by a second number of packets received by the second node during the block period decreased by 1; and
- at step c), the medium time measurement comprises a medium jitter calculated as a difference between the medium one-point reception jitter and the medium one-point transmission jitter.

[0018] Alternatively:

- at step a), the medium transmission time parameter comprises a medium one-point transmission jitter calculated as a difference between a last-packet transmission timestamp indicative of a transmission time of a last packet of the at least two packets and a first-packet transmission timestamp indicative of a transmission time of a first packet of the at least two packets, divided by a first number of packets transmitted by the first node during the block period decreased by 1;
- at step b), the medium reception time parameter comprises a medium one-point reception jitter calculated as a difference between a last-packet reception timestamp indicative of a reception time of the last packet of the at least two packets and a first-packet reception timestamp indicative of a reception time of the first packet of the at least two packets, divided by the first number of packets transmitted by the first node during the block period decreased by 1; and
- at step c), the medium time measurement comprises a medium jitter calculated as a difference between the medium one-point reception jitter and the medium one-point transmission jitter.

[0019] According to preferred embodiments, dividing the packet flow comprises marking the first packets by a first marking value and marking the second packets by a second marking value.

[0020] According to a first embodiment, step a) is performed by the first node and step b) is performed by the second node.

[0021] According to a second alternative embodiment, step a) is performed by a first computer connected to the first node and step b) is performed by a second computer connected to the second node.

[0022] Preferably, step c) is performed by a management server cooperating with the communication network.

[0023] Alternatively, step c) is performed by one of the first node and the second node.

[0024] According to a second aspect, the present invention provides a communication network configured to implement the method as set forth above.

[0025] According to a third aspect, the present invention provides a computer program product, loadable in the memory of at least one computer and including software code portions for performing the steps of the method set forth above, when the product is run on at least one computer.

Brief description of the drawings

[0026] The present invention will become clearer from the following detailed description, given by way of example and not of limitation, to be read with reference to the accompanying drawings, wherein:

- Figure 1 schematically shows an exemplary packet-switched network;
- Figure 2 schematically shows a packet's structure, according to an embodiment of the present invention;
- Figures 3a and 3b are flow charts illustrating the method for performing a time measurement according to an embodiment of the present invention, at the transmitting side;
- Figures 4a and 4b are flow charts illustrating the method for performing a time measurement according to an embodiment of the present invention, at the receiving side; and
- Figures 5a, 5b and 5c show three alternative embodiments of the method for performing a time measurement in the communication network of Figure 1.

Detailed description of preferred embodiments of the invention

[0027] Figure 1 schematically shows an exemplary packet-switched communication network CN in which the method for performing a time measurement according to embodiments of the present invention may be implemented. The communication network CN may be an IP network, an Ethernet network, an MPLS network or any other known type of packet-switched communication network.

[0028] The communication network CN comprises a plurality of nodes reciprocally interconnected by links according to any known topology.

[0029] In particular, the communication network CN comprises a first node N1 and a second node N2. The first node N1 (also termed herein after "transmitting node") is configured to transmit a packet flow PF to the second node (also termed "receiving node"), possibly through intermediate nodes (not shown in Figure 1) of the communication network CN. The transmitting node N1 may be either the source node of the packet flow PF or an intermediate node of the path from the source node to the destination node. Similarly, the receiving node N2 may be either the destination node of the packet flow PF or an intermediate node of the path from the source node to the destination node.

[0030] Preferably, the communication network CN is also suitable for cooperating with a management server MS. The management server MS may be either a stand-alone server connected to any of the nodes of the communication network CN. Alternatively, the management server MS may be implemented at any of the nodes of the communication network CN.

[0031] According to preferred embodiments of the present invention, the packet flow PF comprises a sequence of packets P_{ki} which are marked, i.e. each packet P_{ki} includes a feature (namely, a field comprising at least one bit) whose value is set to one of two alternative marking values V_a, V_b.

[0032] More particularly, as shown in Figure 2, each packet P_{ki} comprises a header H_i and a payload P_i. The payload P_i comprises data to be transmitted. Besides, preferably, the header H_i comprises information for routing the packet P_{ki}, such as the source node address and the destination node address. The packet P_{ki} further comprises a marking field MF, which may be set to one of two alternative marking values V_a, V_b. The marking field MF is preferably comprised in the header H_i. The marking field MF may be for instance a field to which the protocol according to which the packet P_{ki} is formatted has not assigned a specific function yet. Alternatively, the marking field MF may be comprised in a field having other uses. For instance, in case of IP packets, the marking field MF may comprise a bit of the 8-bit DS (Differentiated Service) field, and its two alternative marking values V_a and V_b may be 1 and 0, respectively.

[0033] The packets P_{ki} are preferably marked upon their transmission by the transmitting node N1, which properly sets the value of the marking field MF to one of the two alternative marking values V_a, V_b. The transmitting node N1 periodically changes the marking value V_a, V_b assigned to the marking field MF (for instance, 1 or 0) with a period T_b, which will be termed herein after "block period". This way, packets P_{ki} marked during first block periods (also termed herein after "even block periods") are marked by a first marking value V_a (e.g. 1), while packets P_{ki} marked during second block periods (also termed herein after "odd block periods") which alternate in time with the first block periods are marked by a second marking value V_b (e.g. 0).

[0034] The block period T_b may be set by the network operator, according to the desired time measurement rate (as it will be described in detail herein after, the block period T_b is also the measurement period). For instance, the block period T_b may be equal to 5 minutes.

[0035] According to preferred embodiments of the present invention, at each block period T(k) (k=0, 1, 2, 3, etc.), for each packet P_{ki} a transmission timestamp TS1(i, k) is generated, which indicates the time at which the packet P_{ki} is transmitted by the transmitting node N1. Then, a medium transmission timestamp TSM1(k) of the block period T(k) is preferably calculated according to the following equation:

$$TSM1(k) = \frac{\sum_{i=1}^{C1(k)} TS1(i, k)}{C1(k)}, \quad [3]$$

where C1(k) is the count of packets P_{ki} transmitted by the transmitting node N1 during the block period T(k).

[0036] Further, at each block period T(k), for each packet P_{ki} a reception timestamp TS2(i, k) is generated, which indicates the time at which the packet P_{ki} is received by the receiving node N2. Then, a medium reception timestamp TSM2(k) is preferably calculated according to the following equation:

$$TSM2(k) = \frac{\sum_{i=1}^{C2(k)} TS2(i,k)}{C2(k)}, \quad [4]$$

where $C2(k)$ is the count of packets P_{ki} received by the receiving node $N2$ during the block period $T(k)$. The reception packet count $C2(k)$ is lower than the transmission packet count $C1(k)$ if packets of the packet flow PF were lost during the current block period $T(k)$, while it is equal to the reception packet count $C1(k)$ if no packet loss occurred during the current block period $T(k)$.

[0037] Preferably, for each block period $T(k)$, a medium delay $DM(k)$ is then calculated according to the following equation:

$$DM(k) = TSM2(k) - TSM1(k). \quad [5]$$

[0038] Hence, according to the present invention, instead of providing a delay measurement referring to a single sample packet for each block period, a delay measurement indicative of the average delay undergone by all the packets transmitted during the block period $T(k)$ is provided. In other words, the delay measurement calculated according to the present invention provides an estimate of the performance of the packet flow PF during the block period $T(k)$ as a whole.

[0039] The calculation is advantageously based on a single medium transmission timestamp $TSM1(k)$, which aggregates the transmission timestamps relating to all the packets transmitted during the block period $T(k)$, and a single medium reception timestamp $TSM2(k)$, which aggregates the reception timestamps relating to all the packets received during the block period $T(k)$. Hence, in case the medium delay calculation is carried out by the management server MS , at each block period $T(k)$ each node $N1$ and $N2$ has to transmit a single timestamp $TSM1(k)$ and $TSM2(k)$ to the management server MS , respectively. Hence, the calculation of the medium delay indicative of the delays of all the packets transmitted and received during a block period requires the transmission of two timestamps only through the communication network CN . This advantageously allows saving bandwidth in the communication network CN .

[0040] Furthermore, the medium delay measurement is resistant against possible packet loss and/or reception sequence errors occurring within the block period $T(k)$, as it will be discussed in detail herein after.

[0041] According to a preferred embodiment of the present invention, for each block period $T(k)$, in addition to calculating the medium transmission timestamp $TSM1(k)$ according to equation [3], also a medium one-point transmission jitter $OPJM1(k)$ is calculated according to the following equation:

$$OPJM1(k) = \frac{\sum_{i=1}^{C1(k)-1} TS1(i+1,k) - TS1(i,k)}{C1(k) - 1} = \frac{TS1(C1(k),k) - TS1(1,k)}{C1(k) - 1}. \quad [6]$$

[0042] The medium one-point transmission jitter $OPJM1(k)$ basically is the average of the offsets between transmission times of couples of consecutive packets transmitted during the block period $T(k)$.

[0043] Further, for each block period $T(k)$, in addition to calculating the medium reception timestamp $TSM2(k)$ according to equation [4], also a medium one-point reception jitter $OPJM2(k)$ is calculated according to the following equation:

$$OPJM2(k) = \frac{\sum_{i=1}^{C2(k)-1} TS2(i+1,k) - TS2(i,k)}{C2(k) - 1} = \frac{TS2(C2(k),k) - TS2(1,k)}{C2(k) - 1}. \quad [7]$$

[0044] The medium one-point reception jitter $OPJM2(k)$ basically is the average of the offsets between reception times of couples of consecutive packets received during the block period $T(k)$.

[0045] Preferably, for each block period $T(k)$, a medium jitter $JM(k)$ is then calculated according to the following equation:

$$JM(k) = OPJM2(k) - OPJM1(k). \quad [8]$$

[0046] Hence, according to the present invention, instead of providing a jitter measurement referring to a single couple of sample consecutive packets for each block period, a jitter measurement indicative of the average jitter undergone by all the couples of consecutive packets transmitted during the block period $T(k)$ is provided. Hence, also the jitter measurement calculated according to the present invention provides an estimate of the performance of the packet flow during the block period $T(k)$ as a whole.

[0047] Similarly to the delay measurement described above, also the jitter measurement calculated according to equations [6], [7] and [8] provides the same advantages: reduced amount of information to be transmitted over the communication network CN and resiliency to packet loss and/or reception sequence errors occurring during the block period $T(k)$, as it will be discussed in detail herein after.

[0048] The method for performing time measurements according to the present invention will be now described in further detail.

[0049] By referring first to Figure 3a, at the beginning of each block period $T(k)$, the transmitting node N1 preferably checks whether the block period $T(k)$ is an even block period or an odd block period (step 301), and accordingly determines the marking value V_a (during even block periods) or V_b (during odd block periods) to be applied to packets that will be transmitted during the block period $T(k)$.

[0050] Then, if the block period $T(k)$ is an even block period, two variables are preferably initialised to zero: a first transmission counter $C1a$ and a first cumulative transmission timestamp $TSC1a$ (step 302a).

[0051] Then, the transmitting node N1 waits for possible packets P_{ki} of the packet flow PF to be transmitted (step 303a). The transmitting node N1 preferably identifies the packets P_{ki} of the packet flow PF by using at least part of its destination address, at least part of its source address and, optionally, other fields of the packet header H_i (for instance, the field DSCP in case of IP packets).

[0052] Before transmitting a packet P_{ki} , the node N1 preferably marks it (step 304a) by the marking value V_a . Further, the first transmission counter $C1a$ is preferably increased by 1 (step 305a).

[0053] Then, if the packet P_{ki} is the first packet to be transmitted during the current block period $T(k)$ (step 306a), a first first-packet transmission timestamp $FTS1a$ is preferably set equal to a current time t^* indicated by a local clock of the transmitting node N1 (step 307a). Then, the first cumulative transmission timestamp $TSC1a$ is preferably set equal to the first first-packet transmission timestamp $FTS1a$ (step 308a).

[0054] Otherwise, if the packet P_{ki} is not the first packet to be transmitted during the current block period $T(k)$, a first last-packet transmission timestamp $LTS1a$ is preferably set equal to the current time t^* indicated by the local clock of the transmitting node N1 (step 309a). Then, the first cumulative transmission timestamp $TSC1a$ is preferably increased by the first last-packet transmission timestamp $LTS1a$ (step 310a).

[0055] The packet P_{ki} is then transmitted (step 311a). Step 311a may be carried out substantially in parallel to steps 305a-310a.

[0056] Steps 304a-311a are preferably iterated each time a packet P_{ki} of the packet flow PF has to be transmitted, until the end of the block period $T(k)$ (step 312a).

[0057] If, at step 301, it is determined that the block period $T(k)$ is an odd block period, two variables are preferably initialised to zero: a second transmission counter $C1b$ and a second cumulative transmission timestamp $TSC1b$ (step 302b).

[0058] Then, the transmitting node N1 waits for possible packets P_{ki} to be transmitted (step 303b). Before transmission of a packet P_{ki} , the node N1 preferably marks it (step 304b) by the marking value V_b . Further, the second transmission counter $C1b$ is preferably increased by 1 (step 305b).

[0059] Then, if the packet P_{ki} is the first packet to be transmitted during the current block period $T(k)$ (step 306b), a second first-packet transmission timestamp $FTS1b$ is preferably set equal to the current time t^* indicated by the local clock of the transmitting node N1 (step 307b). Then, the second cumulative transmission timestamp $TSC1b$ is preferably set equal to the second first-packet transmission timestamp $FTS1b$ (step 308b).

[0060] Otherwise, if the packet P_{ki} is not the first packet to be transmitted during the current block period $T(k)$, a second last-packet transmission timestamp $LTS1b$ is preferably set equal to the current time t^* indicated by the local clock of the transmitting node N1 (step 309b). Then, the second cumulative transmission timestamp $TSC1b$ is preferably increased by the second last-packet transmission timestamp $LTS1b$ (step 310b).

[0061] The packet P_{ki} is then transmitted (step 311b). Step 311b may be carried out substantially in parallel to steps 305b-310b.

[0062] Steps 304b-311b are preferably iterated each time a packet P_{ki} of the packet flow PF has to be transmitted, until the end of the block period $T(k)$ (step 312b).

[0063] Referring now to Figure 3b, at the beginning of the successive block period $T(k+1)$, it is checked whether the block period $T(k+1)$ is an even or odd block period (step 320).

[0064] If the block period $T(k+1)$ is an even block period, it is determined that the values of the variables relating to packets marked by V_a (namely $C1a$, $TSC1a$ and $LTS1a$) may be currently changing, whereas the values of the variables relating to packets marked by V_b (namely $C1b$, $TSC1b$ and $LTS1b$) are currently constant. Hence, a transmission packet

count $C1(k)$ is set equal to the value of the second transmission counter $C1b$ reached at the last iteration of step 305b during the previous block period $T(k)$ (step 321a). Further, a medium transmission timestamp $TSM1(k)$ is calculated as the value of the second cumulative transmission timestamp $TSC1b$ divided by the value of the second transmission counter $C1b$ reached at the last iteration of steps 310b and 305b, respectively, during the previous block period $T(k)$ (step 322a). Further, a first-packet transmission timestamp $TS1(1, k)$ is set equal to the value of the second first-packet transmission timestamp $FTS1b$ determined at step 307b during the previous block period $T(k)$ (step 323a). Further, a last-packet transmission timestamp $TS1(C1(k), k)$ is set equal to the value of the second last-packet transmission timestamp $LTS1b$ reached at the last iteration of step 309b during the previous block period $T(k)$ (step 324a).

[0065] Otherwise, if the block period $T(k+1)$ is an odd block period, it is determined that the values of the variables relating to packets marked by Vb (namely $C1b$, $TSC1b$ and $LTS1b$) may be currently changing, whereas the values of the variables relating to packets marked by Va (namely $C1a$, $TSC1a$ and $LTS1a$) are currently constant. Hence, the transmission packet count $C1(k)$ is set equal to the value of the first transmission counter $C1a$ reached at the last iteration of step 305a during the previous block period $T(k)$ (step 321b). Further, the medium transmission timestamp $TSM1(k)$ is calculated as the value of the first cumulative transmission timestamp $TSC1a$ divided by the value of the first transmission counter $C1a$ reached at the last iteration of steps 310a and 305a, respectively, during the previous block period $T(k)$ (step 322b). Further, the first-packet transmission timestamp $TS1(1, k)$ is set equal to the value of the first first-packet transmission timestamp $FTS1a$ determined at step 307a during the previous block period $T(k)$ (step 323b). Further, the last-packet transmission timestamp $TS1(C1(k), k)$ is set equal to the value of the first last-packet transmission timestamp $LTS1a$ reached at the last iteration of step 309a during the previous block period $T(k)$ (step 324b).

[0066] Hence, the transmission packet count $C1(k)$ determined at step 321a or 321b indicates the number of packets Pki transmitted during the block period $T(k)$. Further, the medium transmission timestamp $TSM1(k)$ calculated at step 322a or 322b corresponds to the medium transmission timestamp $TSM1(k)$ calculated according to the above equation [3] and is accordingly a medium transmission timestamp of the packets Pki transmitted during the block period $T(k)$. Further, the first-packet transmission timestamp $TS1(1, k)$ set at step 323a or 323b is the transmission timestamp of the first packet Pki transmitted during the block period $T(k)$. Finally, the last-packet transmission timestamp $TS1(C1(k), k)$ set at step 324a or 324b is the transmission timestamp of the last packet Pki transmitted during the block period $T(k)$.

[0067] Preferably, the time at which steps 312a-324a and 321b-324b are performed is delayed by a safety wait time SWT relative to the start time of the block period $T(k+1)$, for reasons that will be described in detail herein after.

[0068] Then, the transmission packet count $C1(k)$, the first-packet transmission timestamp $TS1(1, k)$, the last-packet transmission timestamp $TS1(C1(k), k)$ and the medium transmission timestamp $TSM1(k)$ are transmitted to the management server MS (step 325), which will use them for calculating a medium delay $DM(k)$ and a medium jitter $JM(k)$ referred to the block period $T(k)$, as it will be described in detail herein after. According to an alternative embodiment, before step 325 is performed, a medium one-point transmission jitter $OPJM1(k)$ may be calculated as the difference between the last-packet transmission timestamp $TS1(C1(k), k)$ and the first-packet transmission timestamp $TS1(1, k)$ divided by the transmission packet count $C1(k)$ decreased by 1. The calculated medium one-point transmission jitter $OPJM1(k)$ corresponds to the medium one-point transmission jitter $OPJM1(k)$ calculated according to the above equation [6]. In such case, at step 325 the medium one-point transmission jitter $OPJM1(k)$ may be transmitted in addition to or instead of $TS1(1, k)$ and $TS1(C1(k), k)$.

[0069] Step 325 may be performed e.g. by establishing a communication session between the transmitting node $N1$ and the management server MS based on a known communication protocol, for instance FTP (File Transfer Protocol). The communication session may be started either by the transmitting node $N1$ or the management server MS .

[0070] Further, step 325 is preferably performed at a random instant comprised between the end of step 324a/324b and the end of the block period $T(k+1)$ brought forward by a further safety wait time SWT' , for reasons that will be described in detail herein after.

[0071] By referring now to Figure 4a, at the beginning of each block period $T(k)$, also the receiving node $N2$ preferably determines whether the block period $T(k)$ is an even block period or an odd block period (step 401).

[0072] Then, if the block period $T(k)$ is an even block period, two variables are preferably initialised to zero: a first reception counter $C2a$ and a first cumulative reception timestamp $TSC2a$ (step 402a).

[0073] Then, the receiving node $N2$ waits for possible packets Pki (step 403a). Also the receiving node $N2$ preferably identifies the packets Pki of the packet flow PF by using at least part of its destination address, at least part of its source address and, optionally, other fields of the packet header Hi (for instance, the field DSCP in case of IP packets).

[0074] Upon reception of a packet Pki , the first reception counter $C2a$ is preferably increased by 1 (step 404a).

[0075] Then, if the packet Pki is the first packet received during the current block period $T(k)$ (step 405a), a first first-packet reception timestamp $FTS2a$ is preferably set equal to a current time t^{**} indicated by a local clock of the receiving node $N2$ (step 406a). Then, the first cumulative reception timestamp $TSC2a$ is preferably set equal to the first first-packet reception timestamp $FTS2a$ (step 407a).

[0076] Otherwise, if the packet Pki is not the first packet received during the current block period $T(k)$, a first last-packet reception timestamp $LTS2a$ is preferably set equal to the current time t^{**} indicated by the local clock of the

receiving node N2 (step 408a). Then, the first cumulative reception timestamp TSC2a is preferably increased by the first last-packet reception timestamp LTS2a (step 409a).

[0077] Steps 404a-409a are preferably iterated each time a packet P_{ki} of the packet flow PF is received, until the end of the block period T(k) (step 410a).

[0078] If, at step 401, it is determined that the block period T(k) is an odd block period, two variables are preferably initialised: a second reception counter C2b and a second cumulative reception timestamp TSC2b (step 402b).

[0079] Then, the receiving node N2 waits for possible packets P_{ki} (step 403b). Upon reception of a packet P_{ki}, the second reception counter C2b is preferably increased by 1 (step 404b).

[0080] Then, if the packet P_{ki} is the first packet received during the current block period T(k) (step 405b), a second first-packet reception timestamp FTS2b is preferably set equal to the current time t** indicated by the local clock of the receiving node N2 (step 406b). Then, the second cumulative reception timestamp TSC2b is preferably set equal to the second first-packet reception timestamp FTS2b (step 407b).

[0081] Otherwise, if the packet P_{ki} is not the first packet received during the current block period T(k), a second last-packet reception timestamp LTS2b is preferably set equal to the current time t** indicated by the local clock of the receiving node N2 (step 408b). Then, the second cumulative reception timestamp TSC2b is preferably increased by the second last-packet reception timestamp LTS2b (step 409b).

[0082] Steps 404b-409b are preferably iterated each time a packet P_{ki} of the packet flow PF is received, until the end of the block period T(k) (step 410b).

[0083] Referring now to Figure 4b, at the beginning of the successive block period T(k+1), it is determined whether the block period T(k+1) is an even or odd block period (step 420).

[0084] If the block period T(k+1) is an even block period, it is determined that the values of the variables relating to packets marked by V_a (namely C2a, TSC2a and LTS2a) may be currently changing, whereas the values of the variables relating to packets marked by V_b (namely C2b, TSC2b and LTS2b) are currently constant. Hence, a reception packet count C2(k) is set equal to the value of the second reception counter C2b reached at the last iteration of step 404b during the previous block period T(k) (step 421a). Further, a medium reception timestamp TSM2(k) is calculated as the value of the second cumulative reception timestamp TSC2b divided by the value of the second reception counter C2b reached at the last iteration of steps 409b and 404b, respectively, during the previous block period T(k) (step 422a). Further, a first-packet reception timestamp TS2(1, k) is set equal to the value of the second first-packet reception timestamp FTS1b determined at step 406b during the previous block period T(k) (step 423a). Further, a last-packet reception timestamp TS2(C2(k), k) is set equal to the value of the second last-packet reception timestamp LTS2b reached at the last iteration of step 408b during the previous block period T(k) (step 424a).

[0085] Otherwise, if the block period T(k+1) is an odd block period, it is determined that the values of the variables relating to packets marked by V_b (namely C2b, TSC2b and LTS2b) may be currently changing, whereas the values of the variables relating to packets marked by V_a (namely C2a, TSC2a and LTS2a) are currently constant. Hence, the reception packet count C2(k) is set equal to the value of the first reception counter C2a reached at the last iteration of step 404a during the previous block period T(k) (step 421b). Further, the medium reception timestamp TSM2(k) is calculated as the value of the first cumulative reception timestamp TSC2a divided by the value of the first reception counter C2a reached at the last iteration of step 409a and 404a, respectively, during the previous block period T(k) (step 422b). Further, the first-packet reception timestamp TS2(1, k) is set equal to the value of the first first-packet reception timestamp FTS2a determined at step 406a during the previous block period T(k) (step 423b). Further, the last-packet reception timestamp TS2(C2(k), k) is set equal to the value of the first last-packet reception timestamp LTS2a reached at the last iteration of step 408a during the previous block period T(k) (step 424b).

[0086] Hence, the reception packet count C2(k) determined at step 421a or 421b indicates the number of packets P_{ki} received during the block period T(k). Further, the medium reception timestamp TSM2(k) calculated at step 422a or 422b corresponds to the medium reception timestamp TSM2(k) calculated according to the above equation [4] and is accordingly a medium reception timestamp of the packets P_{ki} received during the block period T(k). Further, the first-packet reception timestamp TS2(1, k) set at step 423a or 423b is the reception timestamp of the first packet P_{ki} received during the block period T(k). Finally, the last-packet reception timestamp TS2(C2(k), k) set at step 424a or 424b is the reception timestamp of the last packet P_{ki} received during the block period T(k).

[0087] Preferably, the time at which steps 421a-424a and 421b-424b are performed is delayed by a safety wait time SWT relative to the start time of the block period T(k+1). Indeed, due to the propagation delay of the packets P_{ki} through the communication network CN or to a reception sequence error, the last packets marked by V_a transmitted by the transmitting node N1 during an even block period T(k) may be received by the receiving node N2 at the beginning of the subsequent odd block period T(k+1). Hence, at the beginning of an odd block period T(k+1), the variables relating to packets marked by V_a (namely C2a, TSC2a and LTS2a) may still be varying. Similarly, the last packets marked by V_b transmitted by the transmitting node N1 during an odd block period T(k) may be received by the receiving node N2 at the beginning of the subsequent even block period T(k+1). Hence, at the beginning of an even block period T(k+1), the variables relating to packets marked by V_b (namely C2b, TSC2b and LTS2a) may still be varying.

[0088] The safety wait time SWT guarantees that the variables relating to packets marked by Va or Vb are used for calculating the parameters to be transmitted to the management server MS only when their values are stabilized during the block period T(k+1). This also makes the time measurement based on such variables immune from reception sequence errors occurring at the edge between consecutive block periods. The safety wait time SWT is preferably comprised between 1 and 50% of the block period Tb. For instance, if the block period Tb is equal to 5 minutes, the safety wait time SWT may be equal to 20% Tb, namely 1 minute.

[0089] Then, the reception packet count C2(k), the first-packet reception timestamp TS2(1, k), the last-packet reception timestamp TS2(C2(k), k) and the medium reception timestamp TSM2(k) are transmitted to the management server MS (step 425), which will use them for calculating a medium delay DM(k) and a medium jitter JM(k) referred to the block period T(k), as it will be described in detail herein after. According to an alternative embodiment, before step 425 is performed, a medium one-point reception jitter OPJM2(k) may be calculated as the difference between the last-packet reception timestamp TS2(C2(k), k) and the first-packet reception timestamp TS2(1, k) divided by the reception packet count C2(k) decreased by 1. The calculated medium one-point reception jitter OPJM2(k) corresponds to the medium one-point reception jitter OPJM2(k) calculated according to the above equation [7]. In such case, at step 425 the medium one-point reception jitter OPJM2(k) may be transmitted in addition to or instead of TS2(1, k) and TS2(C2(k), k).

[0090] Step 425 may be performed e.g. by establishing a communication session between the receiving node N2 and the management server MS based on a known communication protocol, for instance FTP (File Transfer Protocol). The communication session may be started either by the receiving node N2 or the management server MS.

[0091] Further, step 425 is preferably performed at a random instant comprised between the end of step 424a/424b and the end of the block period T(k+1) brought forwards by a further safety wait time SWT'. This advantageously prevents the management server MS from receiving at the same time the parameters relating to all the nodes of the communication network CN, which may induce a congestion of the management server MS. The further safety wait time SWT' is preferably comprised between 1% and 40% Tb. For instance, if the block period Tb is 5 minutes, the further safety wait time SWT' may be 20%, i.e. 1 minute.

[0092] Hence, at each block period T(k+1), the management server MS receives:

- the transmission packet count C1(k), the first-packet transmission timestamp TS1(1, k), the last-packet transmission timestamp TS1(C1(k), k), the medium transmission timestamp TSM1(k) and, optionally, the medium one-point transmission jitter OPJM1(k) from the transmitting node N1; and
- the reception packet count C2(k), the first-packet reception timestamp TS2(1, k), the last-packet reception timestamp TS2(C2(k), k), the medium reception timestamp TSM2(k) and, optionally, the medium one-point reception jitter OPJM2(k) from the receiving node N2.

[0093] At each block period T(k+1) the management server MS may then calculate the medium delay DM(k) according to the above equation [5], namely as a difference between the medium reception timestamp TSM2(k) and the medium transmission timestamp TSM1(k). The medium delay DM(k) calculated by the management server MS at the block period T(k+1) refers to the preceding block period T(k).

[0094] The medium delay measurement described above is advantageously resistant against possible packet loss and/or reception sequence errors occurring within the block period T(k). Indeed, in case one or more packets are lost before they are received at the receiving node N2, this slightly affects the accuracy of the medium delay calculated according to the equation [5] above, because the two medium timestamps TSM1(k) and TSM2(k) are calculated on different packet counts C1(k), C2(k).

[0095] In particular, in case a single packet is lost during the block period T(k), the error E on the medium transmission timestamp TSM1(k) (and therefore on the medium delay DM(k)) is:

$$E = TSM1(k)^* - TSM1(k), \quad [9a]$$

where TSM1(k)* is the medium transmission timestamp calculated on C1(k)=N-1 packets, namely without taking into account the transmission timestamp of the lost packet, while TSM1(k) is the medium transmission timestamp calculated on C1(K)=N packets, namely taking into account the transmission timestamp of the lost packet. Basically, TSM1(k) is the actually calculated medium transmission timestamp, whereas TSM1(k)* is the medium transmission timestamp which shall be used in equation [5] for providing a correct value of the medium delay DM(k) calculated on N-1 packets. The above equation [9a] may be written also as:

$$E = TSM1(k)^* - \frac{(N-1) \cdot TSM1^*(k) + TS1(i^*, k)}{N} = \frac{TSM1^*(k)}{N} - \frac{TS1(i^*, k)}{N}, \quad [9b]$$

where $TS1(i^*, k)$ is the transmission timestamp of the lost packet.

[0096] Assuming that the transmission timestamps $TS1(i, k)$ are substantially evenly distributed within the current block period $T(k)$, and that accordingly the medium transmission timestamp is close to the middle of the block period $T(k)$, the maximum error E_{max} occurs when the lost packet has the maximum distance from the middle of the block period $T(k)$, namely when the lost packet is either the first packet or the last packet transmitted during the block period $T(k)$. In such cases, its transmission timestamp $TS1(i^*, k)$ is equal to about $TSM1^*(k) \pm T_b/2$, T_b being the duration of the block period. Hence, the maximum error E_{max} may be derived from above equation [9b] as:

$$E_{max} = \frac{TSM1^*(k)}{N} - \frac{TSM1^*(k) \pm T_b/2}{N} = \pm \frac{T_b/2}{N} \quad [9c]$$

[0097] Similarly, the average error E_{av} occurs when the lost packet has an average distance from the middle of the block period $T(k)$, namely when the lost packet has a transmission timestamp $TS1(i^*, k)$ equal to $TSM1^*(k) \pm T_b/4$. Hence, the average error E_{av} may be derived from above equation [9b] as:

$$E_{av} = \frac{TSM1^*(k)}{N} - \frac{TSM1^*(k) \pm T_b/4}{N} = \pm \frac{T_b/4}{N} \quad [9d]$$

[0098] In a more general case where n packets are lost during the block period $T(k)$, the maximum error is $n \cdot E_{max}$ and the average error is $n \cdot E_{av}$, assuming that all the lost packets either precede or follow the middle of the block period $T(k)$ (which is usually the case, because packets are typically lost in bursts).

[0099] From the above equations [9c] and [9d], it is apparent that both the maximum error E_{max} and the average error E_{av} which affects the medium transmission timestamp $TSM1(k)$ (and accordingly also the medium delay $DM(k)$) when packets are lost during the current block period $T(k)$ decrease as the ratio between number of lost packets n and number of transmitted packets N decreases. For instance, in case $T_b=5$ minutes, $n=1$ and $N=10^6$, the maximum error E_{max} is about 0.15 ms.

[0100] Furthermore, the medium delay measurement described above is resistant against possible reception sequence errors occurring during the block period $T(k)$. Indeed, if the order in which packets are received at the receiving node $N2$ is different from the order in which they were transmitted by the transmitting node $N1$, the value of the medium reception timestamp $TSM2(k)$ calculated according to equation [4] does not change. This happens also if, due to a reception sequence error, some packets transmitted during a block period are received during the subsequent block period, as discussed above with reference to the flow chart of Figure 4.

[0101] In addition to the medium delay $DM(k)$, the management server MS may also calculate the medium jitter $JM(k)$ as:

$$JM(k) = \frac{TS2(C2(k), k) - TS2(1, k)}{C2(k) - 1} - \frac{TS1(C1(k), k) - TS1(1, k)}{C1(k) - 1}. \quad [10]$$

[0102] The medium jitter $JM(k)$ calculated according to equation [10] is equal to the medium jitter $JM(k)$ calculated according to above equation [8]. In particular, the first addend of equation [10] is the medium one-point reception jitter $OPJM2(k)$ calculated according to above equation [7] while the second addend of equation [10] is the medium one-point transmission jitter $OPJM1(k)$ calculated according to above equation [6]. Alternatively, in case the management server MS receives from the nodes $N1$, $N2$ the medium one-point jitters $OPJM1(k)$, $OPJM2(k)$, it preferably calculates the medium jitter $JM(k)$ using the above equation [8], namely as their difference. The resulting medium jitter $JM(k)$, in both cases, is the same. The medium jitter $DM(k)$ calculated by the management server MS at the block period $T(k+1)$ refers to the preceding block period $T(k)$.

[0103] According to a particularly preferred embodiment, the management server MS preferably calculates the medium jitter $JM(k)$ according to the following equation:

$$JM(k) = \frac{TS2(C2(k), k) - TS2(1, k)}{C1(k) - 1} - \frac{TS1(C1(k), k) - TS1(1, k)}{C1(k) - 1}. \quad [11]$$

[0104] In other words, both the medium one-point transmission jitter $OPJM1(k)$ and the medium one-point reception jitter $OPJM2(k)$ are calculated on the transmission packet count $C1(k)$. Namely, possible packet loss occurring at the block period $T(k)$ is preferably ignored. This advantageously allows providing an accurate medium jitter measurement

also in case a packet loss occurs during the block period $T(k)$, provided that the packet loss does not involve the first and last packets transmitted during the block period $T(k)$. In any case, even if the packet loss involves the first packet and/or the last packet transmitted during the block period $T(k)$, the error affecting the medium jitter $JM(k)$ calculated according to the above equation [11] decreases as the transmission packet count $C1(k)$ increases.

[0105] The medium jitter $JM(k)$ calculated according to equation [10] or [11] is also resistant against possible reception sequence errors. In particular, reception sequence errors involving intermediate packets transmitted during the block period $T(k)$ do not introduce any error in the medium jitter $JM(k)$, because the result of equations [10] and [11] does not change. Besides, reception sequence errors involving either the first packet or the last packet transmitted during the block period $T(k)$ introduce an error, whose value however increases as the transmission packet count $C1(k)$ increases.

[0106] Hence, delay and jitter measurements indicative of the average delay and jitter undergone by all the packets transmitted during the block period $T(k)$ are provided. In other words, the time measurements calculated according to the present invention provides an estimate of the performance of the packet flow PF during the block period $T(k)$ as a whole.

[0107] The time measurements performed according to embodiments of the present invention advantageously require transmission of few data from the nodes to the management server MS (namely, the medium timestamps for the medium delay calculation and the first/last packet timestamps or the medium one-point jitters for the medium jitter), thereby allowing saving bandwidth in the communication network CN. Furthermore, as discussed above, such time measurements are very resistant against possible packet loss occurring within the block period $T(k)$, especially when the packet counts $C1(k)$, $C2(k)$ in a block period $T(k)$ are very high (1 million or so). Furthermore, as discussed above, such time measurements are very resistant against possible reception sequence errors occurring during the block period $T(k)$ even when, due to a reception sequence error, some packets transmitted during a block period $T(k)$ are received during the subsequent block period $T(k+1)$.

[0108] The time measurement method described above comprises marking the packets P_{ki} of the packet flow PF, and in particular periodically changing the marking value for dividing the packet flow PF in blocks. This is however not limiting. Indeed, according to alternative embodiments of the present invention, the packet flow PF may be divided in blocks without marking the packets P_{ki} . In particular, according to a preferred embodiment, the transmitting node N1 may periodically insert in the packet flow PF an additional packet which marks the edge between a block period $T(k)$ and the subsequent block period $T(k+1)$. For instance, if the communication network CN is an Ethernet network, the additional packets may be OAM frames which, as defined by the ITU-T Recommendation Y.1731 (02/2008), Chapter 8.1 (pages 25-27), are currently used for allowing measurement of the frame loss. In such case, the medium transmission timestamp $TSM1(k)$ relative to the current block period $T(k)$ calculated at the transmitting node N1 may be advantageously transmitted to the receiving node N2 within the OAM packet inserted at the end of the subsequent block period $T(k+1)$.

[0109] In general, in the time measurement method of the present invention, the packet flow PF may be divided in blocks by any technique, provided such technique allows identification of even block periods and odd block periods alternating in time. This allows implementing a double set of variables (counters and timestamps) relating to packets transmitted during even and odd block periods. This way, at each block period, one set of variables is varying (steps of Figures 3a and 4a) while the other set of variables has fixed value and may accordingly be used (steps of Figures 3b and 4b) for calculating the medium timestamps, medium delay and medium jitter as described above.

[0110] Optionally, at the block period $T(k+1)$ the management server MS may also calculate:

- delay of the first packet transmitted/received during the block period $T(k)$ as a difference between the first-packet reception timestamp $TS2(1, k)$ and the first-packet transmission timestamp $TS1(1, k)$;
- delay of the last packet transmitted/received during the block period $T(k)$ as a difference between the last-packet reception timestamp $TS2(C2(k), k)$ and the last-packet transmission timestamp $TS1(C1(k), k)$; and
- jitter between the first packet transmitted/received during the block period $T(k)$ and the last packet transmitted/received during the preceding block period $T(k-1)$ as a difference between the delay of the first packet transmitted/received during the block period $T(k)$ and a delay (calculated at the block period $T(k)$) of the last packet transmitted/received during the preceding block period $T(k-1)$.

[0111] Such calculations provide, for each block period $T(k)$, delay and jitter measurements referring to sample packets, namely the first packet and the last packet of each block period. Such delay and jitter measurements may be compared with the medium delay $DM(k)$ and medium jitter $JM(k)$. In case of a large mismatch, the operator may decide to further investigate for determining the reason why the delay or jitter of the sample packets is so far from the calculated medium values.

[0112] Herein after, a particularly advantageous variant of the method for performing a time measurement according to the present invention will be described in detail.

[0113] With reference again to Figure 3a, at the transmission side, during even (odd) block periods, at step 302a (302b), instead of the cumulative transmission timestamp $TSC1a$ ($TSC1b$), a cumulative transmission time difference $TDC1a$ ($TDC1b$) is provided and initialised to zero. Then, if at step 306a (306b) it is determined that the packet P_{ki} is

the first packet of the block period $T(k)$, only step 307a (307b) is carried out, while step 308a (308b) is preferably omitted. Otherwise, if at step 306a (306b) it is determined that the packet P_{ki} is not the first packet of the block period $T(k)$, step 309a (309b) is carried out as described above, while at step 310a (310b) the value of the cumulative transmission time difference $TDC1a$ ($TDC1b$) is increased by the difference between the first (second) last-packet transmission timestamp $LTS1a$ ($LTS1b$) and the first (second) first-packet transmission timestamp $FTS1a$ ($FTS1b$), namely:

- at step 310a (even block periods): $TDC1a = TDC1a + (LTS1a - FTS1a)$; and
- at step 310b (odd block periods): $TDC1b = TDC1b + (LTS1b - FTS1b)$.

[0114] As step 310a (310b) is iterated, the cumulative transmission time difference $TDC1a$ ($TDC1b$) then cumulates the offsets between the transmission timestamps of the packets transmitted during the block period $T(k)$ and the transmission timestamp of the first packet transmitted during the block period $T(k)$.

[0115] With reference now to Figure 3b, at step 322a (322b) the medium transmission timestamp $TSM1(k)$ may then be calculated as the cumulative transmission time difference $TDC1b$ ($TDC1a$) calculated at the last iteration of step 310b (310a) divided by the second (first) transmission counter $C1b$ ($C1a$), and then increased by the second (first) first-packet transmission timestamp $FTS1b$ ($FTS1a$), namely:

- at step 322a (even block periods):
$$TSM1(k) = \frac{TDC1b}{C1b} + FTS1b ;$$
 and
- at step 322b (odd block periods):
$$TSM1(k) = \frac{TDC1a}{C1a} + FTS1a .$$

[0116] With reference to Figure 4a, at the reception side, during even (odd) block periods, at step 402a (402b), instead of the cumulative reception timestamp $TSC2a$ ($TSC2b$), a cumulative reception time difference $TDC2a$ ($TDC2b$) is provided and initialised to zero. Then, if at step 405a (405b) it is determined that the packet P_{ki} is the first packet of the block period $T(k)$, only step 406a (406b) is carried out, while step 407a (407b) is preferably omitted. Otherwise, if at step 405a (405b) it is determined that the packet P_{ki} is not the first packet of the block period $T(k)$, step 408a (408b) is carried out as described above, while at step 409a (409b) the value of the cumulative reception time difference $TDC2a$ ($TDC2b$) is increased by the difference between the first (second) last-packet reception timestamp $LTS2a$ ($LTS2b$) and the first (second) first-packet reception timestamp $FTS2a$ ($FTS2b$), namely:

- at step 409a (even block periods): $TDC2a = TDC2a + (LTS2a - FTS2a)$; and
- at step 409b (odd block periods): $TDC2b = TDC2b + (LTS2b - FTS2b)$.

[0117] As step 409a (409b) is iterated, the cumulative reception time difference $TDC2a$ ($TDC2b$) then cumulates the offsets between the reception timestamps of the packets received during the block period $T(k)$ and the reception timestamp of the first packet received during the block period $T(k)$.

[0118] With reference now to Figure 4b, at step 422a (422b) the medium reception timestamp $TSM2(k)$ may then be calculated as the second (first) cumulative reception time difference $TDC2b$ ($TDC2a$) calculated at the last iteration of step 409b (409a) divided by the second (first) reception counter $C2b$ ($C2a$), and then increased by the second (first) first-packet reception timestamp $FTS2b$ ($FTS2a$), namely:

- at step 422a (even block periods):
$$TSM2(k) = \frac{TDC2b}{C2b} + FTS2b ;$$
 and
- at step 422b (odd block periods):
$$TSM2(k) = \frac{TDC2a}{C2a} + FTS2a .$$

[0119] In other words, according to such advantageous variant, instead of cumulating the absolute timestamps of the packets transmitted/received in a block period $T(k)$, only their offsets relative to the timestamp of the first packet transmitted/received during the block period $T(k)$ are cumulated. Then, at the subsequent block period $T(k+1)$, the cumulative offset is divided by the number of transmitted/received packets, thereby determining a medium offset which, increased by the timestamp of the first transmitted/received packet, results again in a medium absolute timestamp.

[0120] Advantageously, the cumulative time differences $TDC1a$, $TDC1b$, $TDC2a$, $TDC2b$ have much smaller values than the corresponding cumulative timestamps $TSC1a$, $TSC1b$, $TSC2a$, $TSC2b$. Accordingly, their values may be represented by a much smaller number of digits. The operations 310a, 310b, 409a, 409b are accordingly much easier to

implement from the computational point of view and are less likely to undergo overflow problems.

[0121] In the above description, it has been assumed that all the steps of the flow charts of Figures 3a, 3b are carried out at the transmitting node N1, whereas all the steps of the flow charts of Figures 4a, 4b are carried out at the receiving node N2. This is however not limiting. Indeed, in some cases it may be undesirable or even impossible to implement the method for performing a time measurement at the nodes. This is the case, for instance, when the nodes of the communication network do not comprise the basic functionalities required to support the execution of the method steps, or when the nodes are of different vendors and, accordingly, their reconfiguration would be a very long and costly operation. In such situations, according to alternative embodiments of the present invention, at least some of the method steps are carried out by dedicated computers connected to the nodes, as it will be described in detail herein after.

[0122] According to a first alternative embodiment shown in Figure 5a, the transmitting node N1 is preferably connected to a first computer PC1 through a port different from the port through which the packet flow PF is transmitted. Similarly, the receiving node N2 is preferably connected to a second computer PC2 through a port different from the port through which the packet flow PF is received.

[0123] By referring to the flow charts of Figures 3a and 3b, according to this first alternative embodiment, the transmitting node N1 is configured to:

- mark each packet P_{ki} (steps 301, 303a/303b, 304a/304b);
- create a copy of each marked packet P_{ki} (e.g. by using a mirroring function of the node);
- transmit the packet P_{ki} to the receiving node (step 311a/311b); and
- transmit the copy of the packet P_{ki} to the first computer PC1.

[0124] Further, according to this first alternative embodiment, the first computer PC1 is configured to:

- initialise the variables relating to packets transmitted during the current block period (steps 301 and 302a/302b);
- receive the copies of the marked packets P_{ki} from the transmitting node N1;
- upon reception of each copy, increase the relevant variables (steps 306a/306b to 310a/310b);
- discard the copies of the marked packets P_{ki} received from the transmitting node N1;
- calculate the parameters relating to the previous block period (steps 321a/321b to 324a/324b); and
- transmit the calculated parameters to the management server MS (step 325).

[0125] By referring to the flow charts of Figures 4a and 4b, according to this first alternative embodiment the receiving node N2 is configured to:

- create a copy of each marked packet P_{ki} received from the transmitting node N1 (e.g. by using a mirroring function of the node); and
- transmit the copy of the packet P_{ki} to the second computer PC2.

[0126] Further, according to this first implementation, the second computer PC2 is configured to:

- initialise the variables relating to packets received during the current block period (steps 401 and 402a/402b);
- receive the copies of the marked packets P_{ki} from the receiving node N2;
- upon reception of each copy, increase the relevant variables (steps 404a/404bb to 409a/409b);
- discard the copies of the marked packets P_{ki} received from the receiving node N2;
- calculate the parameters relating to the previous block period (steps 421a/421b to 424a/424b); and
- transmit the calculated parameters to the management server MS (step 425).

[0127] In other words, according to this first implementation, the node N1 only marks the packets P_{ki} for dividing the packet flow PF in blocks, while the time measurement is performed by the computers PC1, PC2 on a copy of the real traffic transmitted in the communication network CN. A marking functionality already available at the node N1 may be exploited for supporting the time measurement performed by the computers PC1, PC2. Advantageously, in case one of the PCs is failed, only the time measurement is affected, whereas transmission of the packet flow PF is unaffected.

[0128] According to a second alternative embodiment shown in Figure 5b, the transmitting node N1 is preferably connected to a first computer PC1 through the same port through which the packet flow PF is transmitted. Similarly, the receiving node N2 is preferably connected to a second computer PC2 through the same port through which the packet flow PF is received. In other words, the first computer PC1 and the second computer PC2 are pass-through probes located on the path followed by the packet flow PF between the transmitting node N1 and the receiving node N2.

[0129] According to this second alternative embodiment, the first computer PC1 is configured to perform all the steps of the flow charts of Figures 3a and 3b, while the second computer PC2 is configured to perform all the steps of the flow

charts of Figures 4a and 4b.

[0130] In other words, according to this second alternative embodiment, the nodes N1, N2 do not perform any operation supporting the time measurement, which is completely in charge to the computers PC1, PC2. The computers PC1, PC2 basically implements all the traffic processing functions of a node, plus the time measurements functions. Hence, the time measurement may be implemented independently of the functionalities available at the nodes N1, N2. This allows implementing the time measurement on real traffic transmitted in communication networks comprising nodes of different types and different vendors.

[0131] According to a third alternative embodiment shown in Figure 5c, the transmitting node N1 is preferably connected to a first computer PC1 from which it receives the traffic flow PF. Besides, the receiving node N2 is preferably connected to a second computer PC2 to which it forwards the traffic flow PF.

[0132] According to this third alternative embodiment, the first computer PC1 is configured to generate the packet flow PF and perform all the steps of the flow charts of Figures 3a and 3b, while the second computer PC2 is configured to perform all the steps of the flow charts of Figures 4a and 4b and terminate the packet flow PF.

[0133] In other words, the computers PC1, PC2 provide an artificial traffic which, between the nodes N1 and N2, is mixed with real traffic exchanged by the nodes N1 and N2. According to such third embodiment, the computers PC1, PC2 then perform the time measurements on the artificial traffic, and accordingly provide results which are approximately indicative of the real traffic performance. Also according to this third alternative embodiment, the nodes N1 and N2 do not perform any operation supporting the time measurement, which is completely in charge to the computers PC1, PC2. Hence, the time measurement may be implemented independently of the functionalities available at the nodes N1, N2. This allows implementing the time measurement on artificial traffic transmitted in communication networks comprising nodes of different types and different vendors.

[0134] The above description is referred to a one-way traffic from N1 to N2. However, the method for performing a time measurement according to embodiments of the present invention may be applied to bidirectional traffic exchanged between N1 and N2. In such case, the node N1 (or the first computer PC1) preferably performs the operations of Figures 3a, 3b on the packet flow transmitted to the node N2, while it performs the operations of Figures 4a, 4b on the opposite packet flow received from the node N2. Symmetrically, the node N2 (or the second computer PC2) preferably performs the operations of Figures 3a, 3b on the packet flow transmitted to the node N1, while it performs the operations of Figures 4a, 4b on the opposite traffic flow received from the node N1. In such case, two medium delays (and jitters) may be calculated, one referring to the propagation direction from N1 to N2 and the other referring to the opposite propagation direction from N2 to N1.

[0135] The medium delays may be advantageously add together, thereby providing a medium round-trip delay between the nodes N1 and N2. Such medium round-trip delay is indeed an estimate, since it is not measured on a same packet transmitted back and forth between N1 and N2. However, it is a quite accurate estimate, since it is calculated as an average on a high number of packets of two counter-propagating packet flows. The inventors have estimated that the accuracy of the medium round-trip delay calculated as described above is comparable to the accuracy of the round-trip delay calculated with a ping function. However, advantageously, differently from the ping function, the calculation of the medium round-trip delay is performed on real traffic and does not require the transmission of any artificial packet.

[0136] Further, in the above description it has been assumed that the nodes N1, N2 transmit the respective calculated parameters (packet count, medium timestamp, first-packet timestamp, last-packet timestamp and, optionally, medium one-point jitter) to the management server MS, which uses them for calculating the medium delay and medium jitter. According to alternative embodiments, the calculation of medium delay and medium jitter may be carried out by one of the nodes N1, N2. Hence, assuming for instance that the calculation is performed by the receiving node N2, the transmitting node N1 shall transmit to the receiving node N1 the calculated parameters. Such parameters may be inserted in a dedicated packet, for instance an OAM frame. Hence, according to such embodiments, the calculation of time measurements relating to various packet flows is advantageously distributed at the nodes of the communication network CN.

Claims

1. A method for performing a time measurement on a packet flow (PF) to be transmitted from a first node (N1) to a second node (N2) of a communication network (CN), said packet flow (PF) being divided, upon its transmission, in first packets transmitted during first block periods and second packets transmitted during second block periods, said first block periods alternating in time with said second block periods, said packet flow (PF) comprising at least two packets (Pki) transmitted by said first node (N1) during a block period (T(k)), with k=0, 1, 2, 3, etc., amongst said first block periods or said second block periods, said method comprising:

a) calculating a medium transmission time parameter (TSM1(k)) for each block period (T(k)), indicative of an

average of at least two transmission time parameters ($TS1(i, k)$) relating to said at least two packets (P_{ki}), wherein said medium transmission time parameter ($TSM1(k)$) comprises a medium transmission timestamp ($TSM1(k)$) calculated as an average of at least two transmission timestamps ($TS1(i, k)$) generated upon transmission of said at least two packets (P_{ki});

b) calculating a medium reception time parameter ($TSM2(k)$) for each block period ($T(k)$), indicative of an average of at least two reception time parameters ($TS2(i, k)$) relating to said at least two packets (P_{ki}), wherein said medium reception time parameter ($TSM2(k)$) comprises a medium reception timestamp ($TSM2(k)$) calculated as an average of at least two reception timestamps ($TS2(i, k)$) generated upon reception of said at least two packets (P_{ki}); and

c) calculating a medium time measurement ($DM(k)$) for each block period ($T(k)$), indicative of an average performance of said packet flow (PF) during said block period ($T(k)$) using said medium transmission time parameter ($TSM1(k)$) and said medium reception time parameter ($TSM2(k)$), wherein said medium time measurement ($DM(k)$) comprises a medium delay ($DM(k)$) calculated as a difference between said medium reception timestamp ($TSM2(k)$) and said medium transmission timestamp ($TSM1(k)$).

2. The method according to claim 1, wherein:

- at said step a), said medium transmission time parameter ($TSM1(k)$, $OPJM1(k)$) comprises a medium one-point transmission jitter ($OPJM1(k)$) calculated as a difference between a last-packet transmission timestamp ($TS1(C1(k), k)$) indicative of a transmission time of a last packet of said at least two packets (P_{ki}) and a first-packet transmission timestamp ($TS1(1, k)$) indicative of a transmission time of a first packet of said at least two packets (P_{ki}), divided by a first number ($C1(k)$) of packets transmitted by said first node ($N1$) during said block period ($T(k)$) decreased by 1;

- at said step b), said medium reception time parameter ($TSM2(k)$, $OPJM2(k)$) comprises a medium one-point reception jitter ($OPJM2(k)$) calculated as a difference between a last-packet reception timestamp ($TS2(C2(k), k)$) indicative of a reception time of said last packet of said at least two packets (P_{ki}) and a first-packet reception timestamp ($TS2(1, k)$) indicative of a reception time of said first packet of said at least two packets (P_{ki}), divided by a second number ($C2(k)$) of packets received by said second node ($N2$) during said block period ($T(k)$) decreased by 1; and

- at said step c), said medium time measurement ($DM(k)$, $JM(k)$) comprises a medium jitter ($JM(k)$) calculated as a difference between said medium one-point reception jitter ($OPJM2(k)$) and said medium one-point transmission jitter ($OPJM1(k)$).

3. The method according to claim 1, wherein:

- at said step a), said medium transmission time parameter ($TSM1(k)$, $OPJM1(k)$) comprises a medium one-point transmission jitter ($OPJM1(k)$) calculated as a difference between a last-packet transmission timestamp ($TS1(C1(k), k)$) indicative of a transmission time of a last packet of said at least two packets (P_{ki}) and a first-packet transmission timestamp ($TS1(1, k)$) indicative of a transmission time of a first packet of said at least two packets (P_{ki}), divided by a first number ($C1(k)$) of packets transmitted by said first node ($N1$) during said block period ($T(k)$) decreased by 1;

- at said step b), said medium reception time parameter ($TSM2(k)$, $OPJM2(k)$) comprises a medium one-point reception jitter ($OPJM2(k)$) calculated as a difference between a last-packet reception timestamp ($TS2(C2(k), k)$) indicative of a reception time of said last packet of said at least two packets (P_{ki}) and a first-packet reception timestamp ($TS2(1, k)$) indicative of a reception time of said first packet of said at least two packets (P_{ki}), divided by said first number ($C1(k)$) of packets transmitted by said first node ($N1$) during said block period ($T(k)$) decreased by 1; and

- at said step c), said medium time measurement ($DM(k)$, $JM(k)$) comprises a medium jitter ($JM(k)$) calculated as a difference between said medium one-point reception jitter ($OPJM2(k)$) and said medium one-point transmission jitter ($OPJM1(k)$).

4. The method according to any of the preceding claims, wherein said dividing comprises marking said first packets by a first marking value (Va) and marking said second packets by a second marking value (Vb).

5. The method according to any of the preceding claims, wherein said step a) is performed by said first node ($N1$) and said step b) is performed by said second node ($N2$).

6. The method according to any of the preceding claims, wherein said step a) is performed by a first computer ($PC1$)

connected to said first node (N1) and said step b) is performed by a second computer (PC2) connected to said second node (N2).

7. The method according to any of the preceding claims, wherein said step c) is performed by a management server (MS) cooperating with said communication network (CN).
8. The method according to any of claims 1 to 6, wherein said step c) is performed by one of said first node (N1) and said second node (N2).
9. A communication network (CN) configured to implement the method according to any of claims 1 to 8.
10. A computer program product, loadable in the memory of at least one computer and including software code portions for performing the steps of the method of any of claims 1 to 8, when the product is run on at least one computer.

Patentansprüche

1. Verfahren zur Ausführung einer Zeitmessung auf einem Paketstrom (PF), der von einem ersten Knoten (N1) zu einem zweiten Knoten (N2) eines Kommunikationsnetzwerks (CN) übertragen wird, wobei der Paketstrom (PF) bei Übertragung in erste Pakete, die während erster Blockzeiten übermittelt werden, und zweite Pakete, die während zweiter Blockzeiten übermittelt werden, unterteilt wird, wobei die ersten Blockzeiten sich in der Zeit mit den zweiten Blockzeiten abwechseln und der Paketstrom (PF) wenigstens zwei Pakete (Pki) umfasst, die durch den ersten Knoten (N1) während einer Blockzeit (T(k)) unter den ersten Blockzeiten oder den zweiten Blockzeiten übertragen werden, wobei $k=0, 1, 2, 3$, usw., das Verfahren umfassend:

- a) Berechnung eines mittleren Übertragungszeitparameters (TSM1(k)) für jede Blockzeit (T(k)), der einen Durchschnitt von wenigstens zwei Übertragungszeitparametern (TS1(i, k)) angibt, die sich auf die wenigstens zwei Pakete (Pki) beziehen, wobei der mittlere Übertragungszeitparameter (TSM1(k)) einen mittleren Übertragungszeitstempel (TSM1(k)) umfasst, der als Durchschnitt von wenigstens zwei Übertragungszeitstempeln (TS1(i, k)) berechnet ist, die bei Übertragung der wenigstens zwei Pakete (Pki) generiert werden;
- b) Berechnung eines mittleren Empfangszeitparameters (TSM2(k)) für jede Blockzeit (T(k)), der einen Durchschnitt von wenigstens zwei Empfangszeitparametern (TS2(i, k)) angibt, die sich auf die wenigstens zwei Pakete (Pki) beziehen, wobei der mittlere Empfangszeitparameter (TSM2(k)) einen mittleren Empfangszeitstempel (TSM2(k)) umfasst, der als Durchschnitt von wenigstens zwei Empfangszeitstempeln (TS2(i, k)) berechnet ist, die bei Empfang der wenigstens zwei Pakete (Pki) generiert werden; und
- c) Berechnung einer mittleren Zeitmessung (DM(k)) für jede Blockzeit (T(k)), die unter Verwendung des mittleren Übertragungszeitparameters (TSM1(k)) und des mittleren Empfangszeitparameters (TSM2(k)) eine durchschnittliche Leistung des Paketstroms (PF) während der Blockzeit (T(k)) angibt, wobei die mittlere Zeitmessung (DM(k)) eine mittlere Verzögerung (DM(k)) umfasst, die als Differenz zwischen dem mittleren Empfangszeitstempel (TSM2(k)) und dem mittleren Übertragungszeitstempel (TSM1(k)) berechnet ist.

2. Verfahren nach Anspruch 1, wobei:

- in dem Schritt a) der mittlere Übertragungszeitparameter (TSM1(k), OPJM1(k)) einen mittleren Einpunktübertragungs-Jitter (OPJM1(k)) umfasst, der als Differenz zwischen einem Übertragungszeitstempel des letzten Pakets (TS1(C1(k), k)), der eine Übertragungszeit eines letzten Pakets der wenigstens zwei Pakete (Pki) angibt, und einem Übertragungszeitstempel des ersten Pakets (TS1(1, k)), der eine Übertragungszeit eines ersten Pakets der wenigstens zwei Pakete (Pki) angibt, geteilt durch eine um 1 verringerte erste Anzahl (C1(k)) von Paketen, die während der Blockzeit (T(k)) durch den ersten Knoten (N1) übertragen wurden, berechnet ist;
- in dem Schritt b) der mittlere Empfangszeitparameter (TSM2(k), OPJM2(k)) einen mittleren Einpunkttempfangs-Jitter (OPJM2(k)) umfasst, der als Differenz zwischen einem Empfangszeitstempel des letzten Pakets (TS2(C2(k), k)), der eine Empfangszeit des letzten Pakets der wenigstens zwei Pakete (Pki) angibt, und einem Empfangszeitstempel des ersten Pakets (TS2(1, k)), der eine Empfangszeit des ersten Pakets der wenigstens zwei Pakete (Pki) angibt, geteilt durch eine um 1 verringerte zweite Anzahl (C2(k)) von Paketen, die während der Blockzeit (T(k)) durch den zweiten Knoten (N2) empfangen wurden, berechnet ist; und
- in dem Schritt c) die mittlere Zeitmessung (DM(k), JM(k)) einen mittleren Jitter (JM(k)) umfasst, der als Differenz zwischen dem mittleren Einpunkttempfangs-Jitter (OPJM2(k)) und dem mittleren Einpunktübertragungs-Jitter (OPJM1(k)) berechnet ist.

3. Verfahren nach Anspruch 1, wobei:

- in dem Schritt a) der mittlere Übertragungszeitparameter ($TSM1(k)$, $OPJM1(k)$) einen mittleren Einpunktübertragungs-Jitter ($OPJM1(k)$) umfasst, der als Differenz zwischen einem Übertragungszeitstempel des letzten Pakets ($TS1(C1(k), k)$), der eine Übertragungszeit eines letzten Pakets der wenigstens zwei Pakete (P_{ki}) angibt, und einem Übertragungszeitstempel des ersten Pakets ($TS1(1, k)$), der eine Übertragungszeit eines ersten Pakets der wenigstens zwei Pakete (P_{ki}) angibt, geteilt durch eine um 1 verringerte erste Anzahl ($C1(k)$) von Paketen, die während der Blockzeit ($T(k)$) durch den ersten Knoten ($N1$) übertragen wurden, berechnet ist;
- in dem Schritt b) der mittlere Empfangszeitparameter ($TSM2(k)$, $OPJM2(k)$) einen mittleren Einpunkttempfangs-Jitter ($OPJM2(k)$) umfasst, der als Differenz zwischen einem Empfangszeitstempel des letzten Pakets ($TS2(C2(k), k)$), der eine Empfangszeit des letzten Pakets der wenigstens zwei Pakete (P_{ki}) angibt, und einem Empfangszeitstempel des ersten Pakets ($TS2(1, k)$), der eine Empfangszeit des ersten Pakets der wenigstens zwei Pakete (P_{ki}) angibt, geteilt durch die um 1 verringerte erste Anzahl ($C1(k)$) von Paketen, die während der Blockzeit ($T(k)$) durch den ersten Knoten ($N1$) übertragen wurden, berechnet ist; und
- in dem Schritt c) die mittlere Zeitmessung ($DM(k)$, $JM(k)$) einen mittleren Jitter ($JM(k)$) umfasst, der als Differenz zwischen dem mittleren Einpunkttempfangs-Jitter ($OPJM2(k)$) und dem mittleren Einpunktübertragungs-Jitter ($OPJM1(k)$) berechnet ist.

4. Verfahren nach einem der vorhergehenden Ansprüche, wobei die Unterteilung die Markierung der ersten Pakete mit einem ersten Markierungswert (V_a) und die Markierung der zweiten Pakete mit einem zweiten Markierungswert (V_b) umfasst.5. Verfahren nach einem der vorhergehenden Ansprüche, wobei der Schritt a) durch den ersten Knoten ($N1$) ausgeführt wird und der Schritt b) durch den zweiten Knoten ($N2$) ausgeführt wird.6. Verfahren nach einem der vorhergehenden Ansprüche, wobei der Schritt a) durch einen ersten Computer ($PC1$) ausgeführt wird, der mit dem ersten Knoten ($N1$) verbunden ist, und der Schritt b) durch einen zweiten Computer ($PC2$) ausgeführt wird, der mit dem zweiten Knoten ($N2$) verbunden ist.7. Verfahren nach einem der vorhergehenden Ansprüche, wobei der Schritt c) durch einen Managementserver (MS) ausgeführt wird, der mit dem Kommunikationsnetzwerk (CN) zusammenarbeitet.8. Verfahren nach einem der Ansprüche 1 bis 6, wobei der Schritt c) entweder durch den ersten Knoten ($N1$) oder den zweiten Knoten ($N2$) ausgeführt wird.9. Kommunikationsnetzwerk (CN), das konfiguriert ist, um das Verfahren nach einem der Ansprüche 1 bis 8 umzusetzen.

10. Computerprogrammprodukt, das in den Speicher von wenigstens einem Computer geladen werden kann, und Softwarecodeanteile für die Durchführung der Schritte des Verfahrens eines der Ansprüche 1 bis 8 enthält, wenn das Produkt auf wenigstens einem Computer ausgeführt wird.

Revendications

1. Procédé de mise en oeuvre d'une mesure de temps sur un flux de paquets (PF) devant être transmis d'un premier noeud ($N1$) à un second noeud ($N2$) d'un réseau de communication (CN), ledit flux de paquets (PF) étant divisé, suite à sa transmission, en des premiers paquets transmis au cours de premières périodes de bloc et des seconds paquets transmis au cours de secondes périodes de bloc, lesdites premières périodes de bloc alternant dans le temps avec lesdites secondes périodes de bloc, ledit flux de paquets (PF) comprenant au moins deux paquets (P_{ki}) transmis par ledit premier noeud ($N1$) au cours d'une période de blocs ($T(k)$), où $k = 0, 1, 2, 3$, etc., parmi lesdites premières périodes de bloc ou lesdites secondes périodes de bloc, ledit procédé comprenant les étapes ci-dessous consistant à :

- a) calculer un paramètre de temps de transmission moyen ($TSM1(k)$) pour chaque période de bloc ($T(k)$), indicatif d'une moyenne d'au moins deux paramètres de temps de transmission ($TS11(i, k)$) connexes auxdits au moins deux paquets (P_{ki}), dans lequel ledit paramètre de temps de transmission moyen ($TSM1(k)$) comprend une estampille temporelle de transmission moyenne ($TSM1(k)$) calculée en tant qu'une moyenne d'au moins

deux estampilles temporelles de transmission ($TS1(i, k)$) générées suite à la transmission desdits au moins deux paquets (P_{ki}) ;

b) calculer un paramètre de temps de réception moyen ($TSM2(k)$) pour chaque période de bloc ($T(k)$), indicatif d'une moyenne d'au moins deux paramètres de temps de réception ($TS2(i, k)$) connexes auxdits au moins deux paquets (P_{ki}), dans lequel ledit paramètre de temps de réception moyen ($TSM2(k)$) comprend une estampille temporelle de réception moyenne ($TSM2(k)$) calculée en tant qu'une moyenne d'au moins deux estampilles temporelles de réception ($TS2(i, k)$) générées suite à la réception desdits au moins deux paquets (P_{ki}) ;

c) calculer une mesure de temps moyenne ($DM(k)$) pour chaque période de bloc ($T(k)$), indicative d'une performance moyenne dudit flux de paquets (PF) au cours de ladite période de bloc ($T(k)$), en utilisant ledit paramètre de temps de transmission moyen ($TSM1(k)$) et ledit paramètre de temps de réception moyen ($TSM2(k)$), dans lequel ladite mesure de temps moyenne ($DM(k)$) comprend un délai moyen ($DM(k)$) calculé en tant qu'une différence entre ladite estampille temporelle de réception moyenne ($TSM2(k)$) et ladite estampille temporelle de transmission moyenne ($TSM1(k)$).

2. Procédé selon la revendication 1, dans lequel :

- à ladite étape a), ledit paramètre de temps de transmission moyen ($TSM1(k)$, $OPJM1(k)$) comprend une gigue de transmission en un point moyenne ($OPJM1(k)$) calculée en tant qu'une différence entre une estampille temporelle de transmission de dernier paquet ($TS1(C1(k), k)$) indicative d'un temps de transmission d'un dernier paquet desdits au moins deux paquets (P_{ki}) et une estampille temporelle de transmission de premier paquet ($TS1(1, k)$) indicative d'un temps de transmission d'un premier paquet desdits au moins deux paquets (P_{ki}), divisée par un premier nombre ($C1(k)$) de paquets transmis par ledit premier noeud ($N1$) au cours de ladite période de bloc ($T(k)$) diminué de 1 ;

- à ladite étape b), ledit paramètre de temps de réception moyen ($TSM2(k)$, $OPJM2(k)$) comprend une gigue de réception en un point moyenne ($OPJM2(k)$) calculée en tant qu'une différence entre une estampille temporelle de réception de dernier paquet ($TS2(C2(k), k)$) indicative d'un temps de réception dudit dernier paquet desdits au moins deux paquets (P_{ki}) et une estampille temporelle de réception de premier paquet ($TS2(1, k)$) indicative d'un temps de réception dudit premier paquet desdits au moins deux paquets (P_{ki}), divisée par un second nombre ($C2(k)$) de paquets reçus par ledit second noeud ($N2$) au cours de ladite période de bloc ($T(k)$) diminué de 1 ; et

- à ladite étape c), ladite mesure de temps moyenne ($DM(k)$, $JM(k)$) comprend une gigue moyenne ($JM(k)$) calculée en tant qu'une différence entre ladite gigue de réception en un point moyenne ($OPJM2(k)$) et ladite gigue de transmission en un point moyenne ($OPJM1(k)$).

3. Procédé selon la revendication 1, dans lequel :

- à ladite étape a), ledit paramètre de temps de transmission moyen ($TSM1(k)$, $OPJM1(k)$) comprend une gigue de transmission en un point moyenne ($OPJM1(k)$) calculée en tant qu'une différence entre une estampille temporelle de transmission de dernier paquet ($TS1(C1(k), k)$) indicative d'un temps de transmission d'un dernier paquet desdits au moins deux paquets (P_{ki}) et une estampille temporelle de transmission de premier paquet ($TS1(1, k)$) indicative d'un temps de transmission d'un premier paquet desdits au moins deux paquets (P_{ki}), divisée par un premier nombre ($C1(k)$) de paquets transmis par ledit premier noeud ($N1$) au cours de ladite période de bloc ($T(k)$) diminué de 1 ;

- à ladite étape b), ledit paramètre de temps de réception moyen ($TSM2(k)$, $OPJM2(k)$) comprend une gigue de réception en un point moyenne ($OPJM2(k)$) calculée en tant qu'une différence entre une estampille temporelle de réception de dernier paquet ($TS2(C2(k), k)$) indicative d'un temps de réception dudit dernier paquet desdits au moins deux paquets (P_{ki}) et une estampille temporelle de réception de premier paquet ($TS2(1, k)$) indicative d'un temps de réception dudit premier paquet desdits au moins deux paquets (P_{ki}), divisée par un premier nombre ($C1(k)$) de paquets transmis par ledit premier noeud ($N1$) au cours de ladite période de bloc ($T(k)$) diminué de 1 ; et

- à ladite étape c), ladite mesure de temps moyenne ($DM(k)$, $JM(k)$) comprend une gigue moyenne ($JM(k)$) calculée en tant qu'une différence entre ladite gigue de réception en un point moyenne ($OPJM2(k)$) et ladite gigue de transmission en un point moyenne ($OPJM1(k)$).

4. Procédé selon l'une quelconque des revendications précédentes, dans lequel ladite étape de division comprend l'étape consistant à marquer lesdits premiers paquets par une première valeur de marquage (V_a) et l'étape consistant à marquer lesdits seconds paquets par une seconde valeur de marquage (V_b).

EP 2 853 062 B1

5. Procédé selon l'une quelconque des revendications précédentes, dans lequel ladite étape a) est mise en oeuvre par ledit premier noeud (N1) et ladite étape b) est mise en oeuvre par ledit second noeud (N2).
- 5 6. Procédé selon l'une quelconque des revendications précédentes, dans lequel ladite étape a) est mise en oeuvre par un premier ordinateur (PC1) connecté audit premier noeud (N1) et ladite étape b) est mise en oeuvre par un second ordinateur (PC2) connecté audit second noeud (N2).
7. Procédé selon l'une quelconque des revendications précédentes, dans lequel ladite étape c) est mise en oeuvre par un serveur de gestion (MS) coopérant avec ledit réseau de communication (CN).
- 10 8. Procédé selon l'une quelconque des revendications 1 à 6, dans lequel ladite étape c) est mise en oeuvre par l'un parmi ledit premier noeud (N1) et ledit second noeud (N2).
9. Réseau de communication (CN) configuré de manière à mettre en oeuvre le procédé selon l'une quelconque des revendications 1 à 8.
- 15 10. Produit-programme informatique, pouvant être chargé dans la mémoire d'au moins un ordinateur et incluant des parties de code logiciel pour mettre en oeuvre les étapes du procédé selon l'une quelconque des revendications 1 à 8, lorsque le produit-programme informatique est exécuté sur au moins un ordinateur.
- 20

25

30

35

40

45

50

55

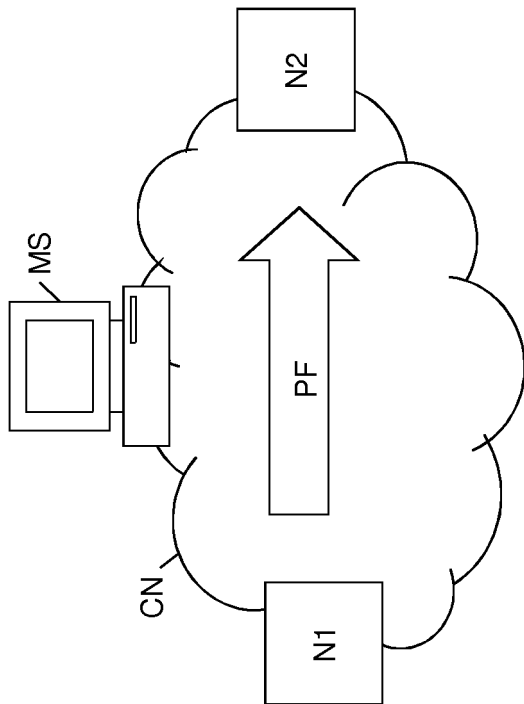


Figure 1

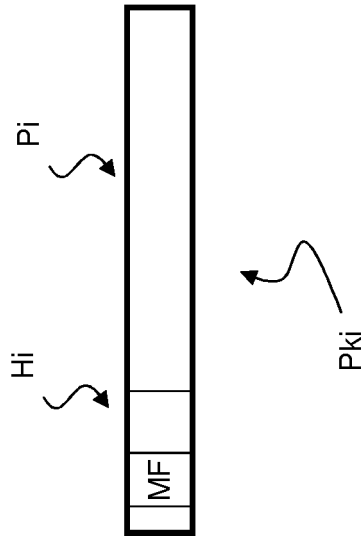


Figure 2

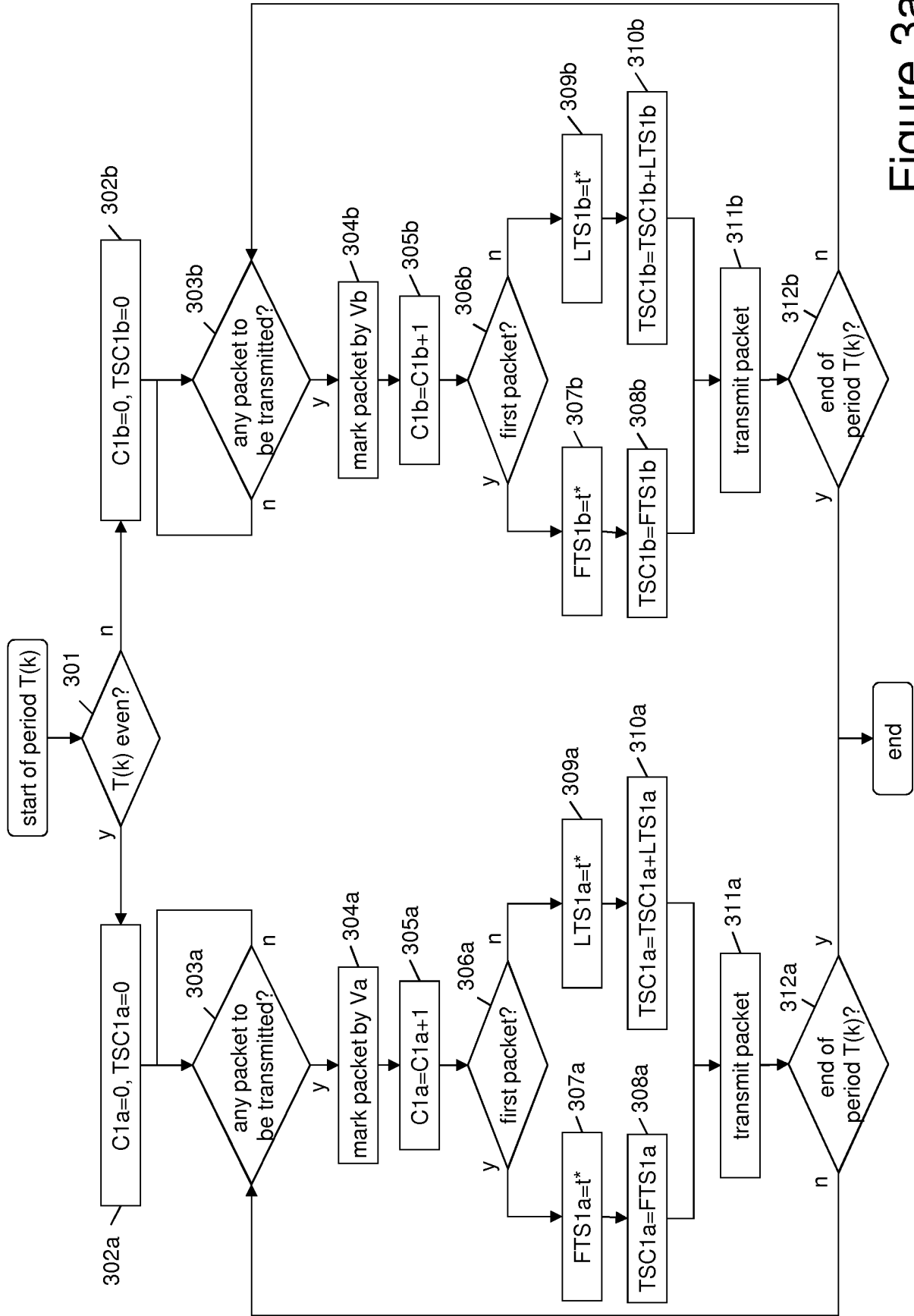


Figure 3a

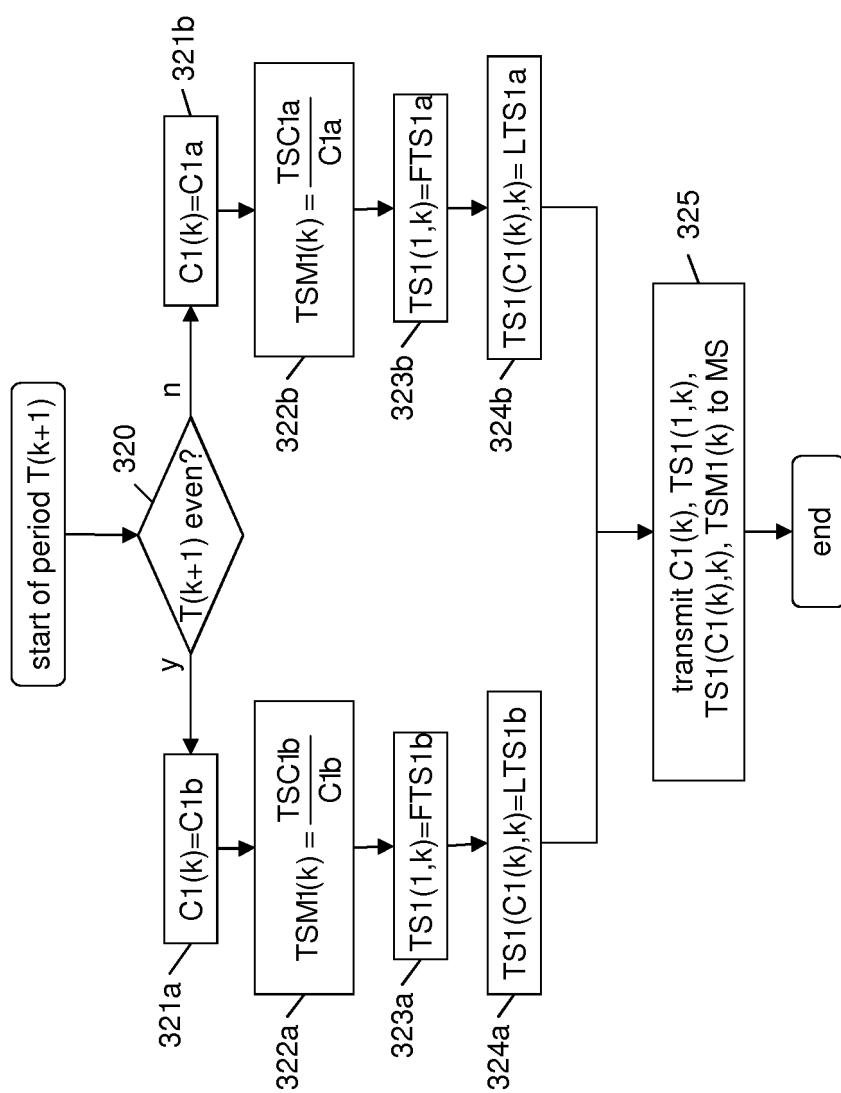


Figure 3b

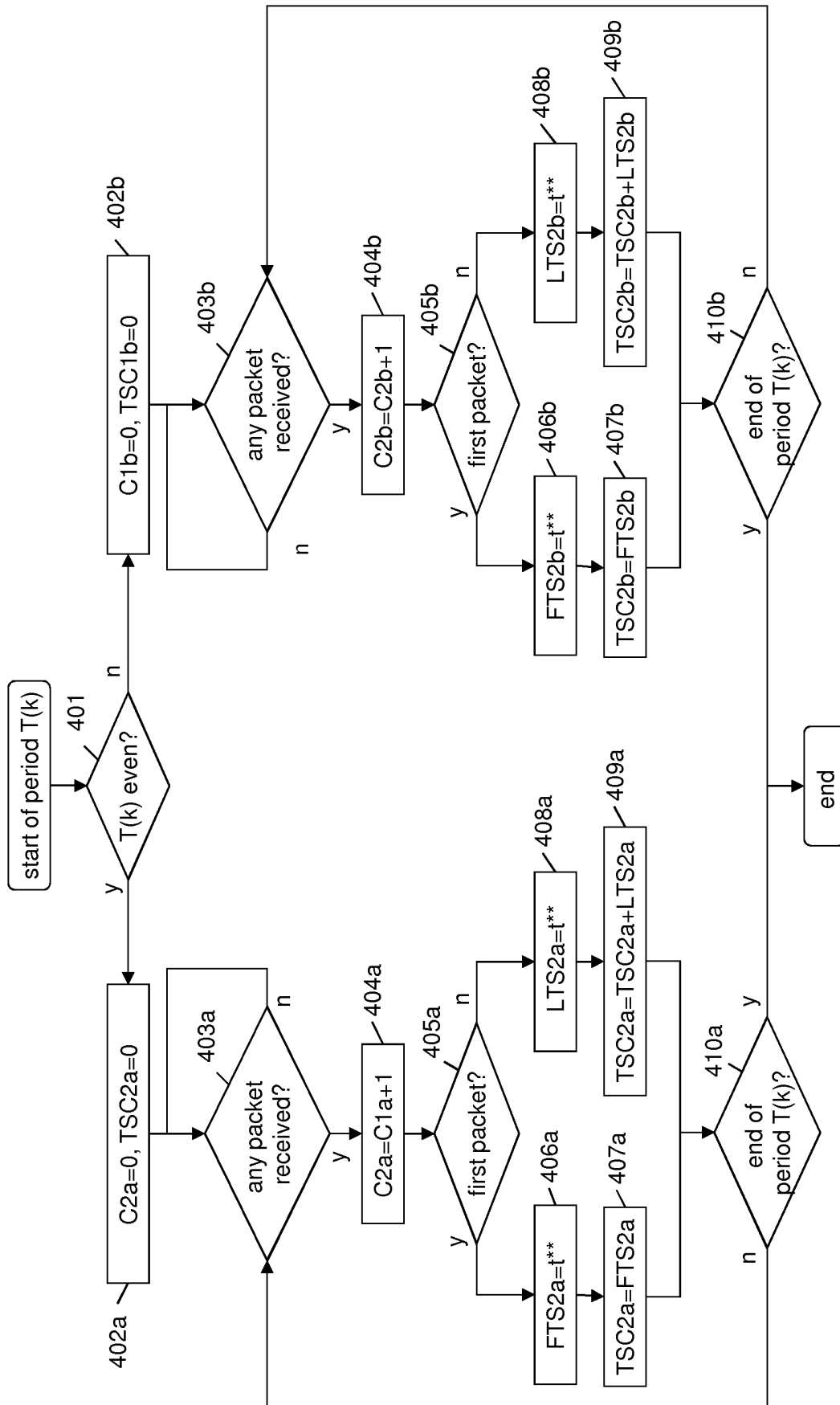


Figure 4a

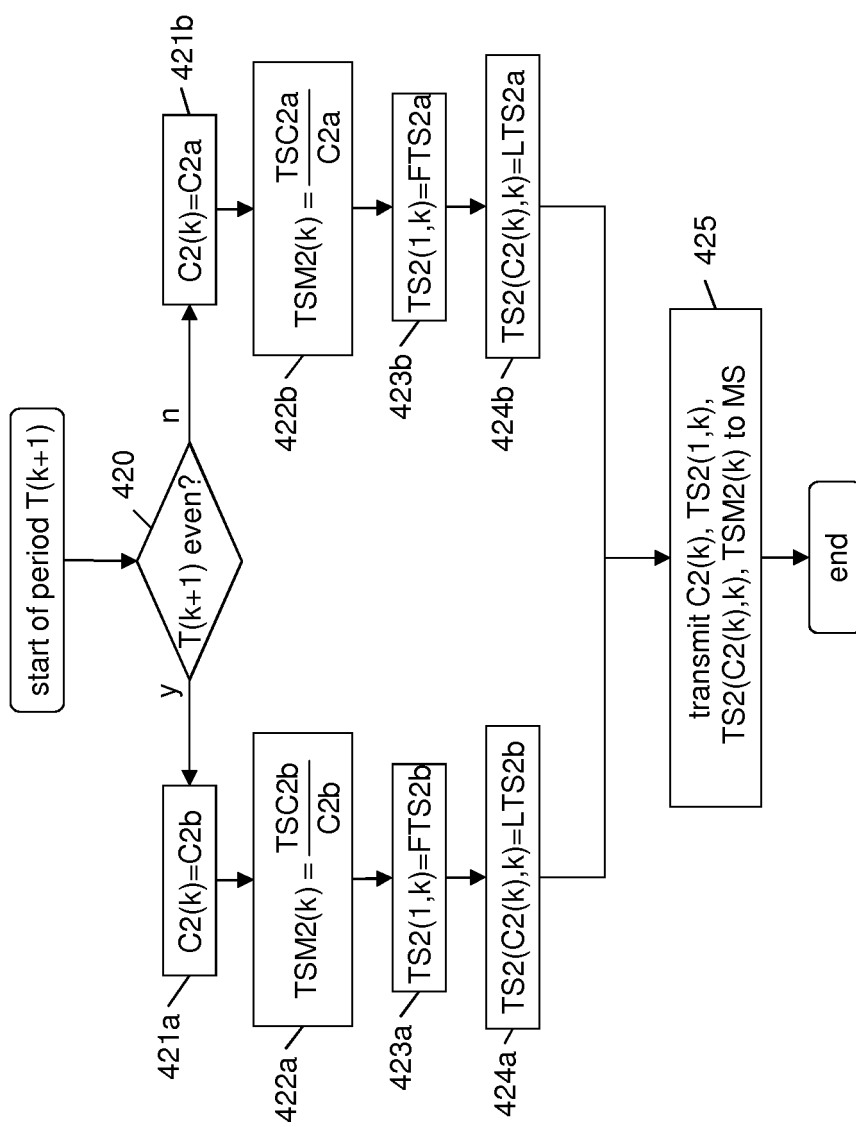


Figure 4b

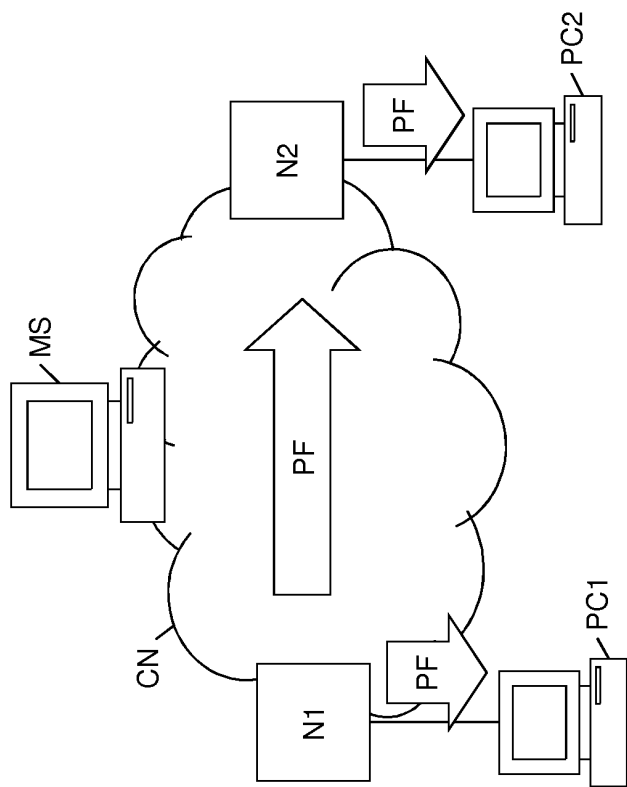


Figure 5a

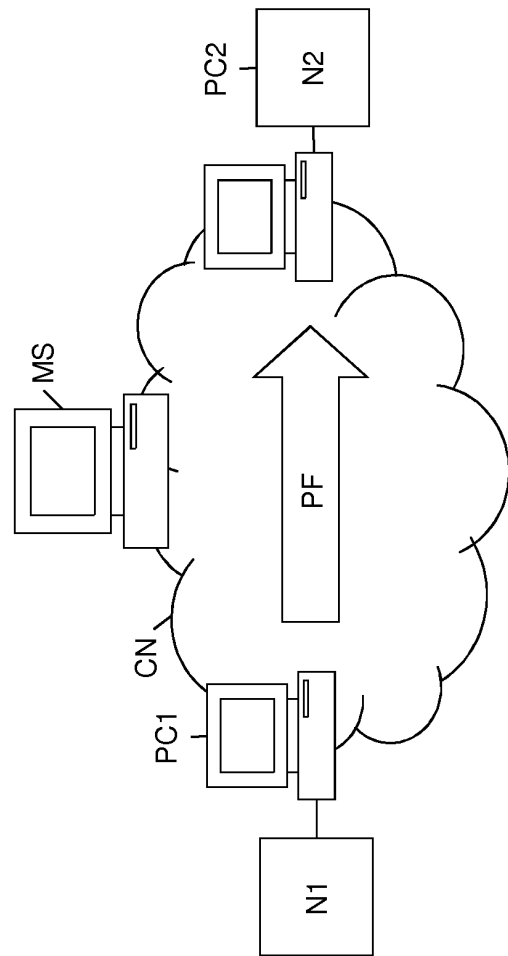


Figure 5b

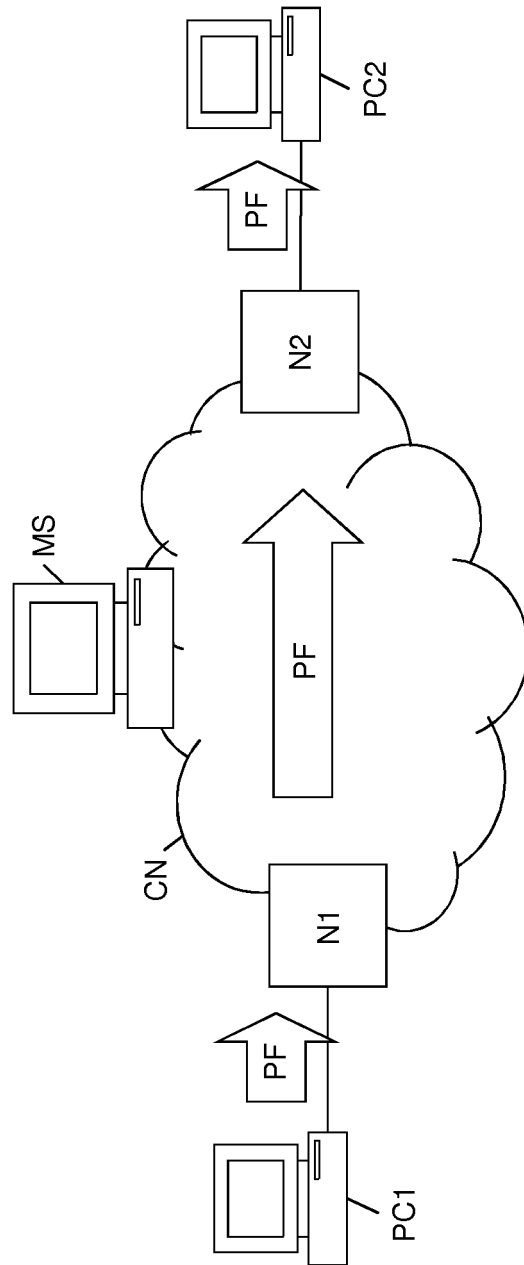


Figure 5c

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- WO 2010072251 A [0008] [0009]
- WO 2011079857 A [0009] [0010]